

## Clase del 3 de Junio

### Variables como argumentos

Se pueden pasar funciones como argumentos a otras funciones

Ejemplo:

```
int sum (int a, int b) {  
    return a + b;  
}  
  
int loop(int c, int x(int, int)) {  
    int acc = 0;  
    for (int i = 0; i < 5; i++) {  
        acc = x(acc, c);  
    }  
    return acc;  
}
```

### Templates

Permiten usar tipos “dinamicos” en funciones:

```
template<typename T>  
void f(T a) {  
    //...  
}  
  
int main () {  
    // Se pasa el tipo al llamarla.  
    f<int>(1);  
    f<string>("abc");  
    return 0;  
}
```

## Clase del 29 de abril

### Estructuras

- struct
- Miembros de una estructura son llamados **campos**.
- **struct** Fecha {  
 int dia;  
 int mes;  
 int anio;

```

}
// Declaración individual.
Fecha f1;
f1.dia = 1;

// Declaración como conjunto.
Fecha f2 = {1, 2, 1970};

```

- Funciones de inicialización:

```

Fecha fecha(ind d, int m, int a) {
    return {d, m, a};
}

```

- Anidamiento (nesting) de estructuras.
- “Funcion de conversión”: funciones para cada estructura que devuelven un string de la misma.
  - Nomenclatura: `structToString` siendo `struct` el nombre de la estructura en minusculas.

## Tipo Abstracto de Dato (TAD)

- Esta compuesto por una estructura y funciones. (Equivalente a metodos)
- Funciones incluidas en el TAD deben comenzar con el nombre de la estructura.

## Sobrecargar funciones

Funcion overloading: redefinir una función con distintos parametros.

Ej.:

```

int sum(int a, int b) {
    return a + b;
}
int sum(float a, float b) {
    return a + b;
}

```

## Referencias

& luego del tipo en la declaración de la función para pasar una referencia. No es necesario derreferenciar variables.

C++:

```

int swap(int& a, int& b) {
    int temp = a;

```

```

    a = b;
    b = temp;
}

```

C:

```

int swap(int *a, int *b) {
    int temp = *a;
    *a = *b;
    *b = temp;
}

```

## Clase del 6 de mayo

### Strings

- \ Character de escape
- Caracteres especiales: \0, \n, \t.
- Strings son terminadas con \0.
- Pueden ser accedidas indexando como array.
- Concatenación de cadenas: a + b.
- Comparación de cadenas: ==, <, >, <=, >=.

### Cosas utiles

- i++, i--, +=, -=, \*=
- for (int i = 0; i < a; i++)

## Clase del 10 de Junio

### Archivos

Conjunto de bytes ordenados, identificados con un nombre, y almacenados en un dispositivo de almacenamiento.

## Clase del 22 de abril

### Temas dados

- bool
- and y or
- Funciones
- Tipos de datos basicos
- Bibliotecas

Variables globales **prohibidas**